**Identified design patterns**

Team member number: 53175

**Patern 1**

Code snippet

Uma imagem com texto

Descrição gerada automaticamente

The exact location on the codebase: class AlwaysWorkingTimeCalenderImpl line 33

Explanation of the rationale for identifying this as a pattern instantiation:

This class create a façade class that implement the interface of GPCalendarCalc and uses the façade to access the subsystem of the class GPCalendarBase.

**Patern 2**

Code snippet

Uma imagem com texto

Descrição gerada automaticamente

The exact location on the codebase: class GPCalendarBase line 34

Explanation of the rationale for identifying this as a pattern instantiation:

In this class GPCalendarBase we can see the pattern factory method.

The class AlwaysWorkingTimeCalendarImpl and GPCalendarBase have a similar order of operations, in the class AlwaysWorkingTimeCalendarImpl we see practical application of generalization and inheritance.

Also, we can notice these two classes sharing similarities like each having a method with a very similar algorithm.

**Patern 3**

Code snippet

Uma imagem com texto

Descrição gerada automaticamente

The exact location on the codebase: class WeekendCalendarImpl, line 60, 62 and 64

Explanation of the rationale for identifying this as a pattern instantiation:

In this class we can see a façade, the variables of the class are objects from the other classes, the class is used as a façade class to access the subsystem.